Journal of Network and Computer Applications ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Push or pull? Toward optimal content delivery using cloud storage ☆

Xinjie Guan *, Baek-Young Choi

University of Missouri-Kansas City, 5110 Rockhill Road, Kansas City, Missouri, MO 64110, USA

ARTICLE INFO

Article history:
Received 17 July 2012
Received in revised form
8 March 2013
Accepted 18 September 2013

Keywords:
Cloud storage
Content delivery

ABSTRACT

Cloud computing and 'Storage As A Service' (SaaS) are experiencing a momentous popularity increase due to its flexible, and scalable access to resources. Especially, cloud storage is becoming an economical alternative to traditional content delivery networks (CDNs) such as Akamai and Limelight Networks for moderate-size content providers. Previous research on content distribution mainly focuses on reducing latency experienced by content customers. A few recent studies address the issue of bandwidth usage in CDNs, as the bandwidth consumption is an important issue due to its relevance to the cost of content providers. However, few researches consider both bandwidth consumption and delay performance for the content providers that use cloud storages with limited budgets, which is the focus of this paper. We develop an efficient light-weight approximation algorithm toward the joint optimization problem of content placement. We also conduct the analysis of its theoretical complexities. The performance bound of the proposed approximation algorithm exhibits a much better worst case than those in previous studies. We further extend the approximate algorithm into a distributed version that allows it to promptly react to dynamic changes in users' interests. The extensive results from both simulations and Planetlab experiments exhibit that the performance is near optimal for most of the practical conditions.

1. Introduction

While traditional Content Distribution Networks (CDNs), such as Akamai and Limelight Networks, can be expensive for moderate-size content providers, and building and managing a CDN infrastructure is becoming increasingly difficult (Broberg et al., 2009), the advent of cloud-based content storage and delivery services provides an economical alternative for those content providers. By outsourcing the tasks of maintaining and delivering a large number of contents to cloud storage providers, content providers, who are also the cloud users, can significantly cut down their expenditures on building and managing a storage infrastructure (Amazon Simple Storage Service (Amazon S3); Broberg et al., 2009; Google Cloud Storage). This economic variation of content placement and delivery attracts a renewed interest on content distribution strategies.

As with traditional CDNs, content providers that use cloud storage are committed to satisfy content users' demands within a reasonable response time. In order to reduce this latency, content providers can disseminate objects on cloud storage servers dispersed in a network near their users. On the other hand, while emphasizing the content users' experience as an overriding concern, content providers also need to consider the expenditure of cloud storage services that is charged on the occupied storage space and traffic volume according to cloud storage providers' polices, such as Amazon Simple Storage Service (S3) and Google Cloud Storage. While replicating objects on cloud servers can lower the cost caused by content delivery traffic by cutting down repetitive transmissions, it, however, raises the cost of additional storage space on cloud servers. This opens a new challenge to design algorithms that could optimize latency as well as cloud storage cost through replicating contents on proper locations.

Various algorithms have been proposed to optimize content delivery that can be mainly categorized as Latency-Minimization (LM) algorithms and Traffic-Minimization (TM) algorithms, according to their optimization aims. The LM algorithms focus on the optimization of latency; while the TM algorithms concern on the optimization of traffic consumed by the delivery of contents in backbone networks. In this paper, we argue that considering the traffic volume together with latency performance under the constraint on storage cost is crucial for economic and efficient content delivery service for content providers using cloud services. We have first formulated the joint traffic-latency optimization problem, and proved its NP-completeness. We then develop an efficient light-weight approximation algorithm, named Traffic-Latency-Minimization (TLM) algorithm, to solve the optimization problem with theoretical provable upper bound for its performance. To limit the frequency of updates to the origin server with local changes such as users interests shift, we also extend our TLM algorithm in a distributed manner. We provide the theoretical analysis for time complexity and space complexity of the TLM algorithm, that are $O(mn \log(n))$, and $O(mn)$ respectively, where m is the number of proxy servers and n is the number of objects. Unlike

☆A part of this paper was published at the IEEE International Conference on Communications (ICC), June 2011.

* Corresponding author. Tel.: +1 816 235 5339.
E-mail addresses: xgck9@mail.umkc.edu, xinjieguan@mail.umkc.edu (X. Guan), choiby@umkc.edu (B.-Y. Choi).

1084-8045/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2013.09.003

most previous works, our algorithm employs fixable and practical conditions that relax many assumptions on parameters such as object size, object request probability, the storage capacity, and the number of requests. Simulation results and experiments show that the performance is near optimal for most of the practical conditions.

The remainder of the paper is organized as follows. Section 2 discusses the background and related work. We formulate our network model and traffic-latency optimization problem, and prove the hardness of the problem in Section 3. We describe our proposed approximation algorithm TLM in both a centralized and a distributed manners, as well as its analysis in Section 4. The performance evaluations and comparisons of TLM with prior algorithms are presented in Section 5. The concluding remarks are given in Section 6.

2. Related work

Content distribution algorithms aim to optimize the system performance with limited resources expressed in various metrics. It is worth noting that those content distribution techniques can be based on a P2P structure as well as on a server/client structure. Hainger and Hartleb (2011) investigated content distribution techniques in both CDNs and P2P networks that are utilized to decrease the traffic load in backbone networks or to optimize content users' experience by shorter end-to-end paths and delays. The motivations of existing content distribution techniques based on CDNs or P2P networks range from improving final users' experience to compressing access cost such as link traffic. Based on the differences on the motivations, most of the content distribution algorithms could be categorized as 'Latency-Minimization' (LM) and 'Traffic-Minimization' (TM).

LM algorithms mainly focus on the optimization of the total communication delay from servers to clients, which is the performance perceived by clients. The average number of autonomous systems (ASes) has been utilized to indicate latency incurred in CDNs in Kangasharju et al. (2002). The authors of Kangasharju et al. (2002) also proposed heuristic algorithms to minimize the average number of ASes traveled for requests. Baev et al. (2008), Baev and Rajaraman (2001), and Korupolu et al. (1999) attempted to reduce clients' access costs for retrieving contents from peers or within the access network. The access cost is related to the distance between content users and replicas (Baev et al., 2008; Korupolu et al., 1999), or it can be a general concept involving all the costs to complete content transmissions (Baev and Rajaraman, 2001). In addition to the communication and access latency, the computational cost is studied and reduced using clustering algorithms in Chen et al. (2003). Chiu and Eun (2010) studied the download latency under a competitive P2P environment, where source peers have a limited capacity of parallel connections. They attempted to achieve minimum download time by dynamically changing the source set of peers under a pull-based model. Similar schemes are employed in grid computing including where distributed resources are shared through a high speed network. In Beck et al. (2002), data are replicated in nearby caches to final user rather than distant source to reduce data transmission time. In Touch (1998), LSAM proxy multicast push web pages to affinity groups for aggregated requests to offload the central server and backbone networks. Moreover, efficient prefetching algorithms are designed in Chu et al. (2007) and Radha et al. (2006) to indicate the most probable disk blocks and push those blocks to user nodes in advance in order to speed up data access.

TM algorithms are designed to lower the traffic volume consumed for contents delivery, so to cut down the expenditure for cloud services. In Almeida et al. (2004), the authors saved the traffic cost through considering the router level and AS level topologies and utilizing multicast streams. Recently, Borst et al. (2010) addressed the issue of reducing the traffic volume for large videos in CDNs. They developed heuristic algorithms for specific topologies by using cache clusters, assuming that many system parameters were constant. The study in Han et al. (2010) adopted various forms of local connectivity and storage for multimedia delivery in a neighbor assisted system to reduce access link traffic. Kamiyama et al. (2011) studied the influence of server allocation in ISP-operated CDNs to the transmission bandwidth consumption and suggested the properties of nodes' topological locations that impact cache placement effectiveness in multiple network topologies. Unified linear programming is utilized to optimal content placement under multiple constraints in Leong et al. (2009). A matrix based *k*-means clustering strategy is proposed in Yuan et al. (2010) to reduce total data movement in scientific cloud workflows. This is where the replication and distribution are constrained by enforcement policies, such as some scientific data are restricted from moving.

Other than the LM and TM algorithms, a few recent works focus on content delivery problems over cloud-based storage. Wei et al. (2010) decreased the storage cost by calculating and maintaining a minimal number of replicas under certain availability requirements. Chaisiri et al. (2009) and Simarro et al. (2011) tried to optimize the content providers' investment by content delivery over multiple cloud storage providers. Broberg et al. (2009) and Lin et al. (2011) designed and implemented frameworks to assist replica placement over cloud storage services, in order to make it possible to optimal content delivery over cloud under diversity requirements.

Our work differs from the previous LM and TM algorithms that we collectively consider content providers' expenditures on traffic volume over cloud storage and content users' experiences. Our aim is to address the optimization problem of traffic consumption and latency for large and diverse sizes of contents by a push–pull hybrid content distribution strategy, where push means replicating objects on certain servers in advance, and pull means only delivering contents that are requested by the content users.

The push–pull strategy has been implemented for content distribution earlier. Zhang et al. (2007) analyzed a P2P pull-based streaming protocol to understand the fundamental limitations and design an effective protocol to achieve better throughput. Feng et al. (2009) investigated theoretic bounds for pull-based protocol under a mesh network and explained the performance gap. The push–pull strategy is also utilized in data aggregation fields to minimize global communication cost in Chakinala et al. (2006). However, those push and pull hybrid protocols are designed for P2P networks or sensor networks without considering the storage of nodes. Therefore, they are not suitable for content distribution over cloud storage where the storage space impacts the content providers investment.

We focus on the environments of cloud based content delivery where the content placement can be actively controlled considering traffic volume while the latency can be controlled under storage constraints. We develop an efficient light-weight approximation algorithm with a provable performance bound, and time and space complexity analysis. We further design a distributed version of the algorithm in which proxy servers can determine object distribution by exchanging local information without requiring global knowledge.

3. Problem formulation

As a traditional content distribution network consists of a central origin server and multiple proxy servers, a content distribution network over a cloud storage is comprised of an origin server and multiple proxy servers on a cloud network. The proxy servers are

Please cite this article as: Guan X, Choi B-Y. Push or pull? Toward optimal content delivery using cloud storage. Journal of Network and Computer Applications (2013), http://dx.doi.org/10.1016/j.jnca.2013.09.003